University of Houston

COSC 6377
Computer Networking

Dr. Omprakash Gnawali

Effect of Instance Location and Network Interface on Computational Performance of a Cluster

Semester Project Milestone 1

Arti Patel
2329771

Ashutosh Kumar
1854393

1 Introduction

In this project, we will use the High-Performance Linpack (HPL) benchmark to find out about the network performance effect on the total performance of a computing cluster.

For our project, we will be using Amazon Web Services (AWS). For the first milestone, we will limit our work to creating a few EC2 instances, setting them up with the appropriate software, and as a bonus, running a test program on them using Message Passing Interface (MPI).

2 Milestone 1 & Workshare

Our work for the first milestone and the workshare division is described in this section.

1. Ashutosh Kumar
   - Created various instances on Amazon Web Services (AWS).
   - Installed OpenBLAS on the various instances.
2. Arti Patel
   - Installed the Message Passing Interface (MPI) implementation (OpenMPI) on the various instances.
   - Downloaded and compiled the HPL benchmark on various instances.

Additionally, as a bonus, we also analysed whether these instances can `ssh` to each other and ran a test program through MPI.

3 Some Details & Challenges

Setting up the EC2 instances on AWS and setting them up with OpenBLAS, OpenMPI, and HPL benchmark was simple enough, however, we had to do some more work before everything worked [1].

We needed the various instances to be able to `ssh` to one another. In order to achieve this, we generated `rsa` public-private keys on each of the instances. The public keys were distributed to the other instances, which gave us the ability to `ssh` to one instance to another. In the 'security group' that these instances belonged to, we added the rule to allow `ssh` communication, which takes place over port 22.

For some reason, we were facing an issue running our test MPI program on one of the instances. Reinstalling OpenMPI fixed that problem.

Another issue that we faced while trying to run the test MPI program across multiple instances was that we did not add the rule to allow `TCP` connections, which is absolutely needed. Once we added this rule, our test program ran swiftly [2].

3.1 Test MPI Program

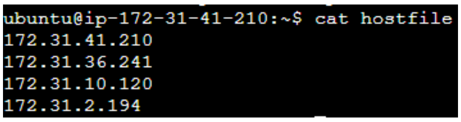
For our test MPI program, we are printing the following from each of our instances [3].

```
Hello world from processor 'IP Address', rank i out of n processors
```

We have created 4 EC2 instances on the same region. The command used to run the MPI program is

```
mpiexec -np 16 --hostfile hostfile ./hello
```

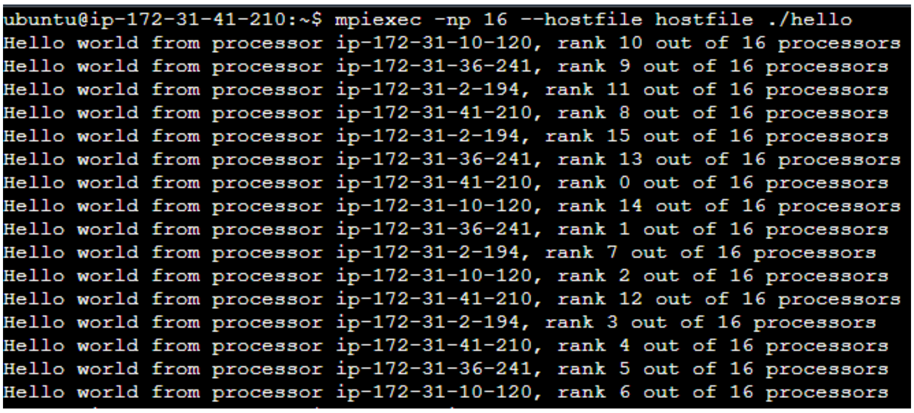
The `hostfile` contains the private IP addresses of the four instances as shown in the following image 1.



```
ubuntu@ip-172-31-41-210:~$ cat hostfile
172.31.41.210
172.31.36.241
172.31.10.120
172.31.2.194
```

Figure 1: Hostfile

The result of the program is given in the following image 2.



```
ubuntu@ip-172-31-41-210:~$ mpiexec -np 16 --hostfile hostfile ./hello
Hello world from processor ip-172-31-10-120, rank 10 out of 16 processors
Hello world from processor ip-172-31-36-241, rank 9 out of 16 processors
Hello world from processor ip-172-31-2-194, rank 11 out of 16 processors
Hello world from processor ip-172-31-41-210, rank 8 out of 16 processors
Hello world from processor ip-172-31-2-194, rank 15 out of 16 processors
Hello world from processor ip-172-31-36-241, rank 13 out of 16 processors
Hello world from processor ip-172-31-41-210, rank 0 out of 16 processors
Hello world from processor ip-172-31-10-120, rank 14 out of 16 processors
Hello world from processor ip-172-31-36-241, rank 1 out of 16 processors
Hello world from processor ip-172-31-2-194, rank 7 out of 16 processors
Hello world from processor ip-172-31-10-120, rank 2 out of 16 processors
Hello world from processor ip-172-31-41-210, rank 12 out of 16 processors
Hello world from processor ip-172-31-2-194, rank 3 out of 16 processors
Hello world from processor ip-172-31-41-210, rank 4 out of 16 processors
Hello world from processor ip-172-31-36-241, rank 5 out of 16 processors
Hello world from processor ip-172-31-10-120, rank 6 out of 16 processors
```

Figure 2: Result of the test MPI program

The above command can be run from any of the four instances.

4 Future Work

For our future work, we will implement more EC2 instances on various instances. Further, we will use the HPL benchmark to test the network performance effect on the total performance of our cluster. We will also use the iPerf tool to measure the highest achievable bandwidth using different interfaces and scenarios

References

[1] How to compile HPL LINPACK on Ubuntu 22.04

[2] Stack Overflow: Unable to run OpenMPI across more than two machines

[3] MPI Hello World Code